Nota: Los ejercicios que tienen (S) son para una "Segunda vuelta" es decir conviene hacerlos una vez que ya se completó la guía haciendo los otros y ya se tiene mas madurez e intuición basica sobre los conceptos. Los que tienen (O) son opcionales por lo cual no se toman en los examenes.

## Procedimientos efectivos

Un concepto importante en ciencias de la computacion es el de *procedimiento* o *metodo* para realizar alguna tarea determinada. Nos interesan los procedimientos que estan definidos en forma precisa e inambigua, es decir aquellos en los cuales en cada paso a seguir, la tarea a realizar esta objetivamente descripta. Ademas siempre supondremos que el interprete o ejecutante es una persona.

Una caracteristica de los procedimientos que surgen en la tarea cientifica es que hay un *conjunto de datos de entrada*, es decir, el conjunto de objetos a partir de los cuales puede comenzar a realizarse el procedimiento. Tambien en los procedimientos que surgen en la tarea cientifica tenemos un *conjunto de datos de salida*, es decir el conjunto de todos los datos que el procedimiento dara como salida en alguna de las posibles ejecuciones al variar los datos de entrada posibles.

Una propiedad importante de los procedimientos es la propiedad de detencion. Si $\mathbb{P}$ es un procedimiento y $d$ es un dato de entrada de $\mathbb{P}$, entonces diremos que $\mathbb{P}$ *se detiene partiendo de* $d$ si en algun momento de la ejecucion de $\mathbb{P}$ partiendo de $d$, el ejecutante concluye una tarea encomendada por $\mathbb{P}$ y no hay una nueva tarea requerida por $\mathbb{P}$ al ejecutante. Puede pasar que para ciertos datos de entrada el procedimiento no se detenga nunca, es decir puede suceder que a medida que se vayan realizando las instrucciones o tareas, siempre el procedimiento direccione a realizar otra tarea especifica y asi sucesivamente.

Cabe destacar que en la gerga computacional a veces se dice "el procedimiento termina" y en realidad a lo que nos referimos es a que termina de realizar la tarea especifica para la cual fue diseñado e inmediatamente se detiene ya que no encomienda mas tareas para hacer. Es decir, en algun sentido terminar es mas fuerte que detenerse ya que el concepto de detencion no presupone la terminacion de ninguna tarea especifica, simplemente se refiere a que el procedimiento llega a un punto en el que no direcciona a ninguna tarea al ejecutante. De todas maneras muchas veces se usara el verbo terminar como sinonimo de detenerse.

Los procedimientos tambien deben ser repetibles, en el sentido de que si realizamos un procedimiento dos veces con el mismo dato de entrada, entonces ambas ejecuciones deben ser identicas, es decir se ralizaran las mismas tareas y en el mismo orden.

Otro aspecto muy importante a considerar es que un procedimiento puede tener pasos a seguir los cuales sean realizables solo en un sentido puramente teorico. Por ejemplo, un procedimiento puede tener una instruccion como la que se muestra a continuacion:

- si el polinomio $ax^5 + bx^4 + 421$ tiene una raiz racional, entonces realizar la tarea descripta en A, en caso contrario realizar la tarea descripta en B

($a, b$ son datos calculados previamente). Como puede notarse mas alla de este aspecto teorico de la instruccion, su descripcion es clara y objetiva, pero en principio no es claro que se pueda ejecutar dicha instruccion en un sentido efectivo a los fines de seguir realizando las siguientes instrucciones. Un procedimiento sera llamado *efectivo* cuando cada paso del mismo sea simple y facil de realizar en forma efectiva por cualquier persona.

En general los procedimientos efectivos que nos interesan son en los que el interprete o ejecutante trabaja solo con papel y lapiz. Ademas el conjunto de datos de entrada siempre estara determinado a priori como parte de la descripcion del procedimiento. Cabe destacar que puede ser muy dificil o imposible, en general, conocer con precision el conjunto de datos de salida de un procedimiento (esto lo justificaremos mas adelante).

Tambien supondremos que los elementos de $\omega$ que intervienen en los datos de entrada y de salida estaran representados por palabras de $Num$ usando la notacion decimal.

Quisas el procedimiento efectivo mas famoso de la matematica es aquel que se enseña en los colegios para sumar dos numeros naturales expresados en notacion decimal. Notar que el conjunto de datos de entrada de dicho procedimiento es $\omega^2$ y el conjunto de datos de salida es el conjunto formado por todas las sumas posibles de pares de elementos de $\omega$, es decir $\omega$. Por supuesto este procedimiento solo usa lapiz, papel y pasos extremadamente simples a seguir en cada momento de la computacion, es decir, en algun sentido, no es necesario "entender que es lo que se esta haciendo" para llegar al final y obtener la palabra que representa en notacion decimal a la suma de los numeros iniciales. Dejamos al lector repasar este procedimiento asi como el que calcula dado un numero $x$ no nulo de $\omega$, al numero $x - 1$, los cuales nos haran falta mas adelante en los ejemplos.

## Funciones $\Sigma$-efectivamente computables

Una funcion $\Sigma$-mixta $f : D_f \subseteq \omega^n \times \Sigma^{*m} \to O$ sera llamada $\Sigma$-*efectivamente computable* si hay un procedimiento efectivo $\mathbb{P}$ tal que

(1) El conjunto de datos de entrada de $\mathbb{P}$ es $\omega^n \times \Sigma^{*m}$

(2) El conjunto de datos de salida esta contenido en $O$.

(3) Si $(\vec{x}, \vec{\alpha}) \in D_f$, entonces $\mathbb{P}$ se detiene partiendo de $(\vec{x}, \vec{\alpha})$, dando como dato de salida $f(\vec{x}, \vec{\alpha})$.

(4) Si $(\vec{x}, \vec{\alpha}) \in (\omega^n \times \Sigma^{*m}) - D_f$, entonces $\mathbb{P}$ no se detiene partiendo desde $(\vec{x}, \vec{\alpha})$

En tal caso diremos que $\mathbb{P}$ *computa* a la funcion $f$.

Veamos algunos ejemplos:

(E1) La funcion $\lambda xy\,[x+y]$ es $\Sigma$-efectivamente computable, cualquiera sea el alfabeto $\Sigma$ ya que el procedimiento enseñado en la escuela primaria para sumar numeros en notacion decimal es efectivo y computa esta funcion.

(E2) La funcion $C_3^{1,2}$ es $\Sigma$-efectivamente computable ya que el siguiente procedimiento $\mathbb{P}$ con conjunto de datos de entrada $\omega \times \Sigma^{*2}$ la computa:

- Independientemente de quien sea el dato de entrada $(x_1, \alpha_1, \alpha_2)$, detenerse dando como salida el numero 3

(E3) La funcion $p_3^{2,3}$ es $\Sigma$-efectivamente computable ya que el siguiente procedimiento con conjunto de datos de entrada $\omega^2 \times \Sigma^{*3}$ la computa:

- Dado el dato de entrada $(x_1, x_2, \alpha_1, \alpha_2, \alpha_3)$, terminar y dar como salida la palabra $\alpha_1$

(E4) *Pred* es $\Sigma$-efectivamente computable. Para realizar el procedimiento efectivo que compute a *Pred* necesitaremos el procedimiento de la escuela primaria que dado un numero no nulo $x$, expresado en notacion decimal, calcula el numero $x-1$, en notacion decimal. Llamemos $\mathbb{P}_{-1}$ a dicho procedimiento. El siguiente procedimiento (con conjunto de datos de entrada igual a $\omega$) computa a *Pred*:

Dado como dato de entrada un elemento $x \in \omega$, realizar lo siguiente:

Etapa 1

Si $x = 0$, entonces ir a Etapa 3, en caso contrario ir a Etapa 2.

Etapa 2

Correr $\mathbb{P}_{-1}$ con dato de entrada $x$ obteniendo $y$ como dato de salida. Dar $y$ como dato de salida y terminar.

Etapa 3

Si $x = 0$, entonces ir a Etapa 1.

Como puede notarse el procedimiento anterior es efectivo ya que debemos entender que en la Etapa 2, los sucesivos pasos efectuados al correr $\mathbb{P}_{-1}$ son todos simples y efectivamente realizables ya que $\mathbb{P}_{-1}$ es efectivo. Por supuesto si uno quisiera ser mas prolijo, deberia reemplazar la Etapa 2 por las distintas instrucciones de $\mathbb{P}_{-1}$, referidas a $x$.

(E5) El predicado $\lambda xy[x < y]$ es $\Sigma$-efectivamente computable cualquiera sea el alfabeto $\Sigma$. Describiremos con palabras un procedimiento que computa a $\lambda xy[x < y]$. Su conjunto de datos de entrada es $\omega^2$. Dado un par $(x, y) \in \omega^2$, el procedimiento primero compara las longitudes de las palabras que en notacion decimal representan a $x$ y $y$. Por supuesto esto lo hace borrando de a un simbolo y viendo si alguna se termina primero. Si resultan de distinta longitud, es facil darse cuenta como sigue. En caso de que las palabras resulten de igual longitud, entonces se hace el procedimiento clasico de ir comparando digitos de izquierda a derecha.

(E6) Veamos que la funcion $\lambda\alpha[|\alpha|]$ es $\Sigma$-efectivamente computable. Como en los lenguajes de programacion, usaremos variables y asignaciones para diseñar el procedimiento. Ademas llamemos $\mathbb{P}_{+1}$ a el procedimiento de la escuela primaria que dado un numero no nulo $x$, expresado en notacion decimal, calcula el numero $x+1$, en notacion decimal. Sea $\mathbb{P}$ el siguiente procedimiento.

Dado como dato de entrada un elemento $\alpha \in \Sigma^*$, realizar lo siguiente:

Etapa 1: Hacer las siguientes asignaciones

$$\begin{array}{rcl} A & \leftarrow & \alpha \\ B & \leftarrow & 0 \end{array}$$

e ir a Etapa 2.

Etapa 2: Si $A$ no es $\varepsilon$, ir a Etapa 3. En caso contrario terminar y dar como salida $B$.

Etapa 3: Correr $\mathbb{P}_{+1}$ con dato de entrada igual al contenido de $B$, obteniendo $y$ como salida. Hacer la asignacion

$$\begin{array}{rcl} A & \leftarrow & \text{resultado de remover el 1er simbolo de } A \\ B & \leftarrow & y \end{array}$$

e ir a Etapa 2.

**Ejercicio 1:** Convensase que el uso de asignaciones puede realizarse usando solo lapiz y papel. Imagine como lo haria en el caso anterior y corrobore que dicho procedimiento es efectivo y ademas computa a $\lambda\alpha[|\alpha|]$.

**Ejercicio 2:** Use los procedimientos $\mathbb{P}_{<}$ y $\mathbb{P}_{|\ |}$ de los dos ejemplos anteriores para diseñar usando asignaciones un procedimiento que compute a la funcion $\lambda i\alpha[[\alpha]_i]$

Un ultimo ejemplo:

(E7) Si $\leq$ es el orden total sobre $\Sigma = \{\blacktriangle, \%\}$ dado por $\blacktriangle < \%$, entonces veremos que la funcion $s^{\leq}$ es $\Sigma$-efectivamente computable. En un par de ejercicios de la Guia2 vimos que cualquiera sea $\alpha \in \Sigma^*$, se cumple

$$\begin{array}{rcl} s^{\leq}(\varepsilon) & = & \blacktriangle \\ s^{\leq}(\alpha\blacktriangle) & = & \alpha\% \\ s^{\leq}(\alpha\%) & = & s^{\leq}(\alpha)\blacktriangle \end{array}$$

y que estas ecuaciones permiten calcular el valor de $s^{\leq}$. Usaremos esta idea para dar un procedimiento efectivo (con conjunto de datos de entrada igual a $\Sigma^*$) que compute a $s^{\leq}$. Como en los lenguajes de programacion, usaremos variables y asignaciones para diseñar el procedimiento.

Etapa 1: Dado el dato de entrada $\alpha \in \Sigma^*$, hacer las siguientes asignaciones

$$\begin{array}{rcl} A & \leftarrow & \alpha \\ B & \leftarrow & \varepsilon \\ F & \leftarrow & \blacktriangle \end{array}$$

e ir a Etapa 2.

Etapa 2: Si $A$ comiensa con $\blacktriangle$, entonces hacer las siguientes asignaciones

$$\begin{array}{rcl} A & \leftarrow & \text{resultado de remover el 1er simbolo de } A \\ F & \leftarrow & B\% \\ B & \leftarrow & B\blacktriangle \end{array}$$

e ir a la Etapa 2. En caso contrario ir a la Etapa 3.

Etapa 3: Si $A$ comiensa con $\%$, entonces hacer las siguientes asignaciones

$$\begin{array}{rcl} A & \leftarrow & \text{resultado de remover el 1er simbolo de } A \\ F & \leftarrow & F\blacktriangle \\ B & \leftarrow & B\% \end{array}$$

e ir a la Etapa 2. En caso contrario ir a la Etapa 4.

Etapa 4: Dar como salida $F$

**Observacion:** Notese que la definicion de funcion $\Sigma$-efectivamente computable para el caso $f = \emptyset$ tiene a priori cierta ambiguedad ya que cualesquiera sean $n, m \in \omega$ y $O \in \{\omega, \Sigma^*\}$ tenemos que $\emptyset : \emptyset \subseteq \omega^n \times \Sigma^{*m} \to O$ ya que $D_\emptyset = \emptyset$ y $I_\emptyset = \emptyset$. De todas maneras, cualesquiera sean los $n, m$ y $O$ elejidos, siempre hay un procedimiento efectivo que computa a $f = \emptyset$, i.e. un procedimiento que nunca se detiene, cualesquiera sea el dato de entrada de $\omega^n \times \Sigma^{*m}$. Es decir que la funcion $\emptyset$ es $\Sigma$-efectivamente computable cualesquiera sea el alfabeto $\Sigma$. Cabe destacar que para el caso de una funcion $f \neq \emptyset$, nuestra definicion es inambigua ya que hay unicos $n, m \in \omega$ y $O \in \{\omega, \Sigma^*\}$ tales que $f : D_f \subseteq \omega^n \times \Sigma^{*m} \to O$.

**Ejercicio 3:** Sea $\mathbb{P}_+$ un procedimiento efectivo que compute a la funcion $\lambda xy\,[x+y]$. Basado en $\mathbb{P}_+$ diseñe un procedimiento efectivo que compute a $\lambda xy\,[x.y]$

**Ejercicio 4:** Sea $\leq$ es el orden total sobre $\Sigma = \{\blacktriangle, \%\}$ dado por $\blacktriangle < \%$. Usando que $s^{\leq}$ es $\Sigma$-efectivamente computable diseñe un procedimiento efectivo que compute a $*^{\leq} : \omega \to \Sigma^*$

**Ejercicio 5:** Sea $\Sigma = \{\blacktriangle, \%\}$ y sea $f : D_f \subseteq \omega \times \omega \times \Sigma^* \rightarrow \omega$ dada por:

$$\begin{array}{rcl} D_f & = & \{(0,1,\varepsilon),(55,54,\blacktriangle\%\blacktriangle\%\blacktriangle),(1,1,\blacktriangle\blacktriangle)\} \\ f(0,1,\varepsilon) & = & 1 \\ f(55,54,\blacktriangle\%\blacktriangle\%\blacktriangle) & = & 2 \\ f(1,1,\blacktriangle\blacktriangle) & = & 3 \end{array}$$

Pruebe que $f$ es $\Sigma$-efectivamente computable.

**Ejercicio 6:** Diga con palabras como diseñaria un procedimiento efectivo que compute la funcion $\lambda ix[(x)_i]$

**Ejercicio 7:** Sean

$$\begin{array}{rcl} f & : & D_f \subseteq \omega \times \omega \times \Sigma^* \rightarrow \omega \\ f_1 & : & D_{f_1} \subseteq \omega \times \Sigma^* \rightarrow \omega \\ f_2 & : & D_{f_2} \subseteq \omega \times \Sigma^* \rightarrow \omega \\ f_3 & : & D_{f_3} \subseteq \omega \times \Sigma^* \rightarrow \Sigma^* \end{array}$$

(a) Es verdad que $D_{f\circ[f_1,f_2,f_3]} = D_{f_1} \cap D_{f_2} \cap D_{f_3}$?

(b) De una descripcion del dominio de $f \circ [f_1, f_2, f_3]$.

(c) Si $f, f_1, f_2, f_3$ son $\Sigma$-efectivamente computables, entonces $f\circ[f_1, f_2, f_3]$ lo es.

**Ejercicio 8:** V o F o I. Justifique

(a) Si $\mathbb{P}$ es un procedimiento efectivo que computa una funcion no vacia $f : D_f \subseteq \omega \rightarrow \omega$ entonces el conjunto de datos de salida de $\mathbb{P}$ es $\omega$

(b) Si $\mathbb{P}$ y $\mathbb{Q}$ son procedimientos efectivos, entonces $\mathbb{PQ}$ lo es.

(c) Denotemos con $e$ a la cantidad de veces que estornudó Alan Turing a lo largo de su vida. Sea $f : \omega \rightarrow \omega$, dada por $f(x) = x^e$. Entonces $f$ es $\emptyset$-efectivamente computable.

(d) Si $f$ y $g$ son $\Sigma$-efectivamente computables entonces $f.g$ lo es

## Conjuntos $\Sigma$-efectivamente computables

Sea $X$ un conjunto cualquiera y sea $S \subseteq X$. Usaremos $\chi_S^X$ para denotar la funcion

$$\begin{array}{rcl} \chi_S^X : X & \rightarrow & \omega \\ x & \rightarrow & \left\{ \begin{array}{l} 1 \text{ si } x \in S \\ 0 \text{ si } x \notin S \end{array} \right. \end{array}$$

Llamaremos a $\chi_S^X$ la *funcion caracteristica de $S$ con respecto a $X$.*

Un conjunto $S \subseteq \omega^n \times \Sigma^{*m}$ sera llamado $\Sigma$-*efectivamente computable* cuando la funcion $\chi_S^{\omega^n \times \Sigma^{*m}}$ sea $\Sigma$-efectivamente computable. Notese que $S$ es $\Sigma$-efectivamente computable sii hay un procedimiento efectivo $\mathbb{P}$, el cual computa $\chi_S^{\omega^n \times \Sigma^{*m}}$, es decir:

- El conjunto de datos de entrada de $\mathbb{P}$ es $\omega^n \times \Sigma^{*m}$, siempre termina y da como dato de salida un elemento de $\{0, 1\}$.

- Dado $(\vec{x}, \vec{\alpha}) \in \omega^n \times \Sigma^{*m}$, $\mathbb{P}$ da como salida al numero 1 si $(\vec{x}, \vec{\alpha}) \in S$ y al numero 0 si $(\vec{x}, \vec{\alpha}) \notin S$.

Si $\mathbb{P}$ es un procedimiento efectivo el cual computa a $\chi_S^{\omega^n \times \Sigma^{*m}}$, diremos que $\mathbb{P}$ *decide la pertenecia a* $S$, con respecto al conjunto $\omega^n \times \Sigma^{*m}$.

**Observacion:** Notese que la definicion de conjunto $\Sigma$-efectivamente computable, para el caso $S = \emptyset$, tiene a priori cierta ambiguedad ya que cualesquiera sean $n, m \in \omega$ tenemos que $\emptyset \subseteq \omega^n \times \Sigma^{*m}$. De todas maneras, cualesquiera sean los $n, m$ elejidos, siempre hay un procedimiento efectivo que computa a $\chi_\emptyset^{\omega^n \times \Sigma^{*m}}$, i.e. un procedimiento que siempre da como salida 0, cualesquiera sea el dato de entrada de $\omega^n \times \Sigma^{*m}$. Es decir que el conjunto $\emptyset$ es $\Sigma$-efectivamente computable cualesquiera sea el alfabeto $\Sigma$. Cabe destacar que para el caso de un conjunto $S \neq \emptyset$, nuestra definicion es inambigua ya que hay unicos $n, m \in \omega$ tales que $S \subseteq \omega^n \times \Sigma^{*m}$.

**Ejercicio 9:** $\omega^3 \times \Sigma^{*2}$ es $\Sigma$-efectivamente computable

**Ejercicio 10:** $\{(x, \alpha) \in \omega \times \Sigma^* : |\alpha|^x \text{ es par}\}$ es $\Sigma$-efectivamente computable

**Ejercicio 11:** Sea $\Sigma = \{\blacktriangle, \%\}$ y sea

$$S = \{(0, 1, \varepsilon), (55, 54, \blacktriangle\%\blacktriangle\%\blacktriangle), (1, 1, \blacktriangle\blacktriangle)\}$$

Pruebe que $S$ es $\Sigma$-efectivamente computable.

**Ejercicio 12:** Sean $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$ conjuntos $\Sigma$-efectivamente computables. Entonces $S_1 \cup S_2, S_1 \cap S_2$ y $(\omega^n \times \Sigma^{*m}) - S_1$ son $\Sigma$-efectivamente computables.

**Ejercicio 13:** (S) Sean $S \subseteq \omega$ y $L \subseteq \{@, \uparrow\}^*$ tales que $S \times L$ es $\{@, \uparrow\}$-efectivamente computable y $(0, \varepsilon) \in S \times L$. Pruebe que entonces $L$ es $\{@, \uparrow\}$-efectivamente computable

## Conjuntos $\Sigma$-efectivamente enumerables

Supongamos que $k, l, n, m \in \omega$ y que $F : D_F \subseteq \omega^k \times \Sigma^{*l} \rightarrow \omega^n \times \Sigma^{*m}$. Supongamos ademas que $n + m \geq 1$. Entonces denotaremos con $F_{(i)}$ a la funcion $p_i^{n,m} \circ F$. Notar que

$$\begin{aligned} F_{(i)} &: D_F \subseteq \omega^k \times \Sigma^{*l} \rightarrow \omega, \text{ para cada } i = 1, ..., n \\ F_{(i)} &: D_F \subseteq \omega^k \times \Sigma^{*l} \rightarrow \Sigma^*, \text{ para cada } i = n+1, ..., n+m \end{aligned}$$

Por lo cual cada una de las funciones $F_{(i)}$ son $\Sigma$-mixtas.

**Ejercicio 14:** Sea $F : \{(x, \alpha) \in \omega \times \Sigma^* : |\alpha|^x \text{ es par}\} \subseteq \omega \times \Sigma^* \rightarrow \omega^2 \times \Sigma^*$ dada por $F(x, \alpha) = (x, x^2 + |\alpha|, \varepsilon)$. Diga quienes son $F_{(1)}$,$F_{(2)}$ y $F_{(3)}$. Que funcion es $[F_{(1)}, F_{(2)}, F_{(3)}]$?

**Ejercicio 15:** En la definicion anterior, para el caso $n + m = 1$, quien es $F_{(1)}$?

**Ejercicio 16:** Pruebe que $F = [F_{(1)}, ..., F_{(n+m)}]$

Un conjunto $S \subseteq \omega^n \times \Sigma^{*m}$ sera llamado $\Sigma$-*efectivamente enumerable* cuando sea vacio o haya una funcion $F : \omega \rightarrow \omega^n \times \Sigma^{*m}$ tal que $I_F = S$ y $F_{(i)}$ sea $\Sigma$-efectivamente computable, para cada $i \in \{1, ..., n + m\}$.

**Observacion:** Notese que para el caso $n = m = 0$, la condicion anterior de que $F_{(i)}$ sea $\Sigma$-efectivamente computable, para cada $i \in \{1, ..., n + m\}$ se cumple vacuamente y por lo tanto la definicion anterior nos dice que un conjunto $S \subseteq \omega^0 \times \Sigma^{*0} = \{\lozenge\}$ sera $\Sigma$-efectivamente enumerable sii es vacio o hay una funcion $F : \omega \rightarrow \{\lozenge\}$, tal que $I_F = S$. Por supuesto, esto nos dice que $\emptyset$ y $\{\lozenge\}$ son $\Sigma$-efectivamente enumerables.

Para entender mejor la idea de esta definicion consideremos el siguiente resultado.

**Lemma 1** *Un conjunto no vacio $S \subseteq \omega^n \times \Sigma^{*m}$ es $\Sigma$-efectivamente enumerable sii hay un procedimiento efectivo $\mathbb{P}$ tal que*

*(1) El conjunto de datos de entrada de $\mathbb{P}$ es $\omega$*

*(2) $\mathbb{P}$ se detiene para cada $x \in \omega$*

*(3) El conjunto de datos de salida de $\mathbb{P}$ es igual a $S$. (Es decir, siempre que $\mathbb{P}$ se detiene, da como salida un elemento de $S$, y para cada elemento $(\vec{x}, \vec{\alpha}) \in S$, hay un $x \in \omega$ tal que $\mathbb{P}$ da como salida a $(\vec{x}, \vec{\alpha})$ cuando lo corremos con $x$ como dato de entrada)*

**Ejercicio 17:** (S) Explique con palabras una justificacion del lema anterior

Cuando un procedimiento $\mathbb{P}$ cumpla (1), (2) y (3) del lema anterior, diremos que $\mathbb{P}$ *enumera* a $S$. O sea que $S$ es $\Sigma$-efectivamente enumerable sii es vacio o hay un procedimiento efectivo el cual enumera a $S$.

Dicho de otra forma un conjunto no vacio $S$ es $\Sigma$-efectivamente enumerable sii hay un procedimiento efectivo $\mathbb{P}$ el cual tiene conjunto de datos de entrada $\omega$ y ademas para los sucesivos datos de entrada $0, 1, 2, 3, ...$, el procedimiento $\mathbb{P}$ produce respectivamente los datos de salida $e_0, e_1, e_2, e_3, ...$ de manera que $S = \{e_0, e_1, e_2, ...\}$. Cabe destacar aqui que puede suceder que $e_i = e_j$, para ciertos $i, j$, con $i \neq j$.

**Nota Importante:** en lo que sigue muchas veces daremos procedimientos que son efectivos en terminos de otros que ya se han dado, es decir daremos un procedimiento que en principio no es claro que sea efectivo pero el cual se volveria efectivo si reemplazamos ciertas instrucciones por la manera efectiva de simularlas. Para hacer mas dinamico el discurso no distinguiremos entre este tipo de procedimientos (efectivisables) y los efectivos propiamente dichos.

Algunos ejemplos:

(E1) Un procedimiento efectivo que enumera $\omega \times \omega$ es el siguiente:

Etapa 1

Si $x = 0$, dar como salida el par $(0, 0)$ y terminar. Si $x \neq 0$, calcular $x_1 = (x)_1$ y $x_2 = (x)_2$.

Etapa 2

Dar como dato de salida el par $(x_1, x_2)$ y terminar

Como puede notarse el procedimiento es efectivo (efectivisable) y ademas el conjunto de datos de salida es $\omega \times \omega$ ya que si tomamos un par cualquiera $(a, b) \in \omega \times \omega$, el procedimiento lo dara como dato de salida para la entrada $x = 2^a 3^b$.

(E2) Veamos que $\omega^2 \times \Sigma^{*3}$ es $\Sigma$-efectivamente enumerable cualquiera sea el alfabeto no vacio $\Sigma$. Sea $\leq$ un orden total para el alfabeto $\Sigma$. Utilisando el orden $\leq$ podemos diseñar el siguiente procedimiento para enumerar $\omega^2 \times \Sigma^{*3}$:

Etapa 1

Si $x = 0$, dar como salida $(0, 0, \varepsilon, \varepsilon, \varepsilon)$ y terminar. Si $x \neq 0$, calcular

$x_1 = (x)_1$

$x_2 = (x)_2$

$\alpha_1 = *^{\leq}((x)_3)$

$\alpha_2 = *^{\leq}((x)_4)$

$\alpha_3 = *^{\leq}((x)_5)$

Etapa 2

Dar como dato de salida la 5-upla $(x_1, x_2, \alpha_1, \alpha_2, \alpha_3)$.

**Ejercicio 18:** Explique por que es efectivo el procedimiento de (E2)

**Ejercicio 19:** Pruebe que el procedimiento de (E2) tiene conjunto de datos de salida igual a $\omega^2 \times \Sigma^{*3}$.

**Ejercicio 20:** Sea $S = \{(x, \uparrow^x) : x \text{ es par}\}$. Pruebe que $S$ es $\{@, \uparrow\}$-efectivamente enumerable.

**Ejercicio 21:** Sean $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$ conjuntos $\Sigma$-efectivamente enumerables. Entonces $S_1 \cup S_2$ es $\Sigma$-efectivamente enumerable

**Ejercicio 22:** (S) Sean $S_1, S_2 \subseteq \omega^n \times \Sigma^{*m}$ conjuntos $\Sigma$-efectivamente enumerables. Entonces $S_1 \cap S_2$ es $\Sigma$-efectivamente enumerable. (No es facil, en el apunte esta probado en forma de lema.)

**Ejercicio 23:** (Explicado en video en granlogico.com) Si $\mathbb{P}$ es un procedimiento efectivo cuyo conjunto de datos de entrada es $\omega \times \Sigma^*$ entonces el conjunto $\{(x, \alpha) \in \omega \times \Sigma^* : \mathbb{P} \text{ termina partiendo de } (x, \alpha)\}$ es $\Sigma$-efectivamente enumerable. Saque como conclusion que el dominio de una funcion $\Sigma$-efectivamente computable es $\Sigma$-efectivamente enumerable.

**Lemma 2** *Si $S \subseteq \omega^n \times \Sigma^{*m}$ es $\Sigma$-efectivamente computable entonces $S$ es $\Sigma$-efectivamente enumerable.*

**Proof.** Supongamos $S \neq \emptyset$. Sea $(\vec{z}, \gamma) \in S$, fijo. Sea $\mathbb{P}$ un procedimiento efectivo que compute a $\chi_S^{\omega^n \times \Sigma^{*m}}$. Ya vimos en el ejemplo anterior que $\omega^2 \times \Sigma^{*3}$ es $\Sigma$-efectivamente enumerable. En forma similar se puede ver que $\omega^n \times \Sigma^{*m}$ lo es. Sea $\mathbb{P}_1$ un procedimiento efectivo que enumere a $\omega^n \times \Sigma^{*m}$. Entonces el siguiente procedimiento enumera a $S$:

Etapa 1

Realizar $\mathbb{P}_1$ con $x$ de entrada para obtener como salida un $(\vec{x}, \vec{\alpha}) \in \omega^n \times \Sigma^{*m}$.

Etapa 2

Realizar $\mathbb{P}$ con $(\vec{x}, \vec{\alpha})$ de entrada para obtener el valor Booleano $e$ de salida.

Etapa 3

Si $e = 1$ dar como dato de salida $(\vec{x}, \vec{\alpha})$. Si $e = 0$ dar como dato de salida $(\vec{z}, \gamma)$.

■

**Ejercicio 24:** Sea $\Sigma = \{@, !, \%\}$. Supongamos $f : S \subseteq \omega \times \Sigma^* \to \Sigma^*$ es $\Sigma$-efectivamente computable. Suponga ademas que $(1, @@) \in S$ y $f(1, @@) = @!!$. Pruebe que el conjunto

$$\{(x, \alpha) \in S : f(x, \alpha) = @!!\}$$

es $\Sigma$-efectivamente enumerable.

**Ejercicio 25:** Sea $S = \{(x, \alpha) \in \omega \times \{@, \uparrow\}^* : \alpha$ tiene exactamente $x$ ocurrencias de $@\}$. Pruebe que $S$ es $\{@, \uparrow\}$-efectivamente enumerable.

**Ejercicio 26:** (Explicado en video en granlogico.com) Si $S \subseteq \omega$ y $f : S \to \omega$ es $\Sigma$-efectivamente computable, entonces el conjunto

$$\{x \in S : x \text{ es par } x/2 \in S \text{ y } f(x) = f(x/2)\}$$

es $\Sigma$-efectivamente enumerable.

**Ejercicio 27:** Hacer

(a) (S) Si $S \subseteq \omega^2 \times \{@, \uparrow\}^{*3}$ es finito, entonces es $\{@, \uparrow\}$-efectivamente computable

(b) (S) Si $f : D_f \subseteq \omega \times \omega \times \{@, \uparrow\}^* \to \omega$ es tal que $D_f$ es finito, entonces $f$ es $\{@, \uparrow\}$-efectivamente computable

**Ejercicio 28:** V o F o I. Justifique

(a) Denotemos con $e$ a la cantidad de veces que estornudó Alan Turing a lo largo de su vida. Sea $S = \{e\}$. Entonces $S$ es un conjunto $\emptyset$-efectivamente computable

(b) $(2, 4, 6, 8, 10, ...)$ es $\Sigma$-efectivamente enumerable

(c) Si $\mathbb{P}$ es un procedimiento efectivo, entonces $I_{\mathbb{P}}$ es $\Sigma$-efectivamente enumerable

(d) Sea $S \subseteq \omega$ y supongamos $\mathbb{P}$ es un procedimiento efectivo el cual enumera a $S$. Entonces el siguiente procedimiento (con dato de entrada $x \in \omega$) enumera a $\{u \in S : u \text{ es par}\}$:

   i. Etapa 1: Guardar en la variable $X$ el valor $x$.
   ii. Etapa 2: Correr $\mathbb{P}$ con dato de entrada $X$ y guardar en la variable $E$ el dato de salida dado por $\mathbb{P}$
   iii. Etapa 3: Si $E$ es par, dar como salida $E$ y detenerse. Caso contrario aumentar en 1 el valor de $X$ e ir a Etapa 2